What is Claimed is:

1. A computer-implemented method, comprising:

executing a client module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a server module that models behavior of a component of the electronic system; and

returning process control to the client module after execution of the server module.

2. The method of claim 1, wherein the client module is a Verilog/PLI module.

3. The method of claim 1, wherein the server module is a System C module.

4. The method of claim 1, wherein the RPC uses the TCP protocol as a transport layer protocol.

5. The method of claim 1, wherein the RPC uses the UDP protocol as a transport layer protocol.

6. The method of claim 1, further comprising mapping a plurality of input ports of the server module to a plurality of signals.

7. The method of claim 1, further comprising suspending operation of the server module.

8. The method of claim 1, further comprising returning a return value to the client module after execution of the server module, the return value representing a plurality of output signals.

9. The method of claim 1, further comprising advancing simulation time by one cycle of a clock signal.

10. The method of claim 9, wherein the server module is configured to be sensitive to a positive edge of the clock signal.

11. A computer-readable medium having computer-executable instructions for:

executing a client module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a server module that models behavior of a component of the electronic system; and

returning process control to the client module after execution of the server module.

12. The computer-readable medium of claim 11, wherein the client module is a Verilog/PLI module.

13. The computer-readable medium of claim 11, wherein the server module is a SystemC module.

14. The computer-readable medium of claim 11, wherein the RPC uses the TCP protocol as a transport layer protocol.

15. The computer-readable medium of claim 11, wherein the RPC uses the UDP protocol as a transport layer protocol.

16. The computer-readable medium of claim 11, having further computer-executable instructions for mapping a plurality of input ports of the server module to a plurality of signals.

17. The computer-readable medium of claim 11, having further computer-executable instructions for suspending operation of the server module.

18. The computer-readable medium of claim 11, having further computer-executable instructions for returning a return value to the client module after execution of the server module, the return value representing a plurality of output signals.

19. The computer-readable medium of claim 11, having further computer-executable instructions for advancing simulation time by one cycle of a clock signal.

20. The computer-readable medium of claim 19, wherein the server module is configured to be sensitive to a positive edge of the clock signal.

21. A computer-implemented method, comprising:

executing a Verilog/PLI module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a SystemC module that models behavior of a component of the electronic system;

suspending operation of the SystemC module;

advancing simulation time by one cycle of a clock signal; and

returning a return value to the Verilog/PLI module after execution of the SystemC module, the return value representing a plurality of output signals.

22. A computer-readable medium having computer-executable instructions for:

executing a Verilog/PLI module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a SystemC module that models behavior of a component of the electronic system;

suspending operation of the SystemC module;

advancing simulation time by one cycle of a clock signal; and

returning a return value to the Verilog/PLI module after execution of the SystemC module, the return value representing a plurality of output signals.

23. A computer-implemented method, comprising:

executing a Verilog/PLI module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a SystemC module that models behavior of a component of the electronic system, the System C module having a plurality of global signals mapped to at least one of an input port of the System C module and an output port of the System C module;

suspending operation of the System C module;

advancing simulation time by one cycle of a clock signal having a 50% duty cycle; and

returning a pointer associated with a return value to the Verilog/PLI module after execution of the System C module, the return value representing a plurality of output signals.

24. The method of claim 23, further comprising generating the System C module by modifying a model created using the C programming language.

25. The method of claim 23, further comprising implementing the RPC at least in part in the Verilog/PLI module.

26. The method of claim 23, further comprising implementing the RPC at least in part in the System C module.

27. A computer-readable medium having computer-executable instructions for:

executing a Verilog/PLI module configured to simulate behavior of an electronic system;

using a remote procedure call (RPC) to transfer process control to a System C module that models behavior of a component of the electronic system, the System C module having a plurality of global signals mapped to at least one of an input port of the System C module and an output port of the System C module;

suspending operation of the System C module;

advancing simulation time by one cycle of a clock signal having a 50% duty cycle; and

returning a pointer associated with a return value to the Verilog/PLI module after execution of the System C module, the return value representing a plurality of output signals.

28. The computer-readable medium of claim 27, having further computer-executable instructions for generating the System C module by modifying a model created using the C programming language.

29. The computer-readable medium of claim 27, having further computer-executable instructions for implementing the RPC at least in part in the Verilog/PLI module.

30. The computer-readable medium of claim 27, having further computer-executable instructions for implementing the RPC at least in part in the System C module.